

UNIVERSITY OF MICHIGAN

EECS 270: Intro to Logic Design

Final Exam

Prof. Karem A Sakallah

Tuesday December 14, 2021
4:00-6:00 p.m.

A - L: 2505 GGBL || M - Z: 1571 GGBL

Name: _____
UMID: _____
Honor Pledge: “I have neither given nor received aid on this exam, nor have I concealed any violation of the Honor Code.”
Signature: _____

Instructions:

- The exam is closed book except for **three** 8.5"x11" sheets of notes. No electronics of any kind may be used.
- Print your name and student ID number and sign the honor pledge.
- Make sure your answers and meaningful work are on the pages with numbers at the bottom. We will be scanning and looking at only these pages: all work on the backs of pages will **not** be checked for determining partial credit.
- The exam consists of 9 problems with the point distribution indicated here. Please keep this in mind as you work through the exam. Use your time wisely.
- There are 10 pages in this exam. Make sure that you have all 10 pages and notify an instructor if you do not.

1.	/10
2.	/8
3.	/12
4.	/10
5.	/15
6.	/15
7.	/5
8.	/15
9.	/10
Total:	/100

1 [Boolean Algebra Basics—10 points]

$2^n -$

- a. An n -variable switching function with \underline{m} distinct maxterms in its canonical POS form has $2^n - 2^m$ distinct minterms in its canonical SOP form.

$$\begin{array}{ccccc} & 0 & & 1 & \\ \text{D} & 0 & & 0 & \\ & 0 & & 1 & \\ & & & | & \\ & & & 6 & \end{array}$$

- b. A NOR gate can be used to check the equivalence of two bits.

$$a \oplus b = a'b + ab'$$

- c. $a \oplus b = a \odot b'$.

$$a \odot b = a'b + ab$$

$$\begin{array}{ccc} x & y & z \\ 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{array}$$

- d. $xy + (zx + zy) = xy + (z\bar{x} \oplus zy)$.

$$\begin{array}{ccc} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{array}$$

- e. The number of possible switching functions of $\underline{20}$ variables is 2^{20} .

$$0111 \quad a'bc \quad \underbrace{d}_{atb' + c' + d}$$

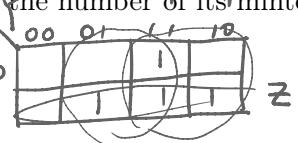
- f. $m'_7(a, b, c, d) = a' + b \cancel{+} c + d$.

- g. An n -variable OR function has $\underline{1}$ maxterm.

$$\begin{array}{c} y \\ \backslash \\ x \end{array}$$

A switching function is *balanced* if the number of its minterms is equal to the number of its maxterms.

- h. $xy + zx + zy$ is balanced.



- i. $a + b + c$ is balanced.

- j. $a \oplus b \oplus c$ is balanced.

0 0 0	0
0 0 1	1
0 1 0	1
0 1 1	0
1 0 0	0
1 0 1	0
1 1 0	1
1 1 1	1

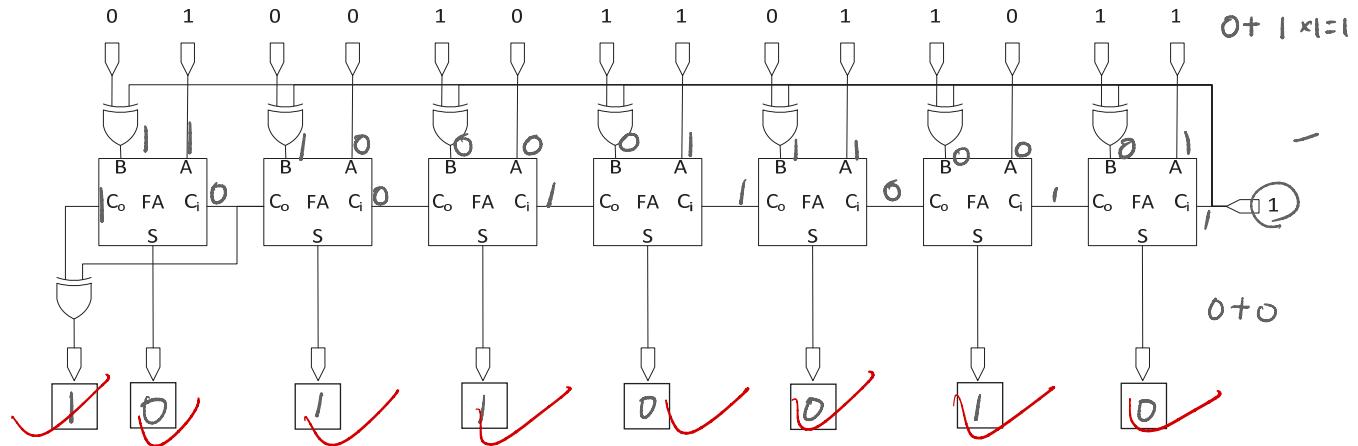
True False

2 [Adder/Subtractor–8 points]

- a. [4 points] The figure below shows a 7-bit ripple-carry adder/subtractor with a specified bit pattern on its input pins. Determine the corresponding bit pattern on its outputs and write it out in the boxes corresponding to the respective output pins.

$$C_o = AB + (A \oplus B)C_i$$

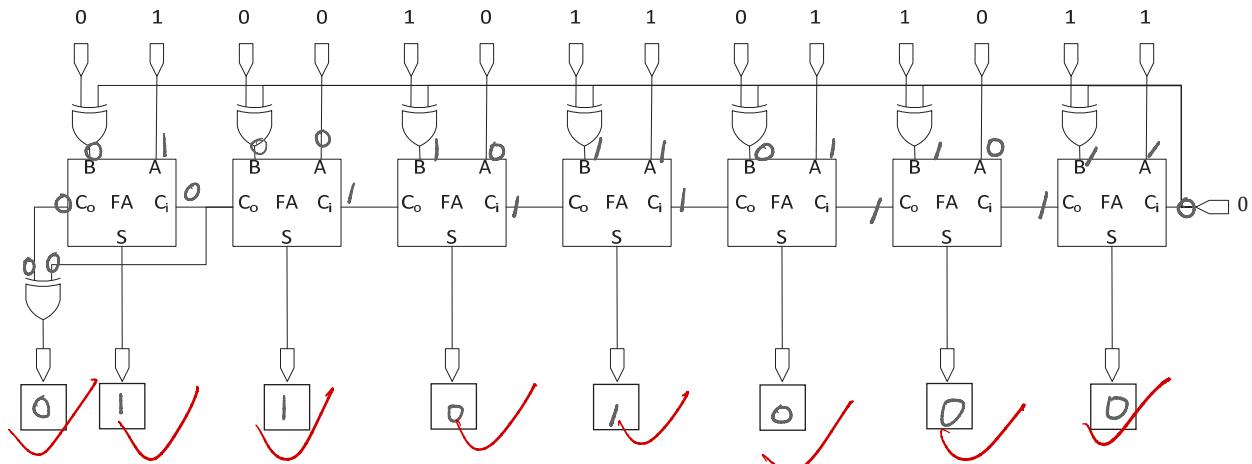
$$S = A \oplus B \oplus C_i$$



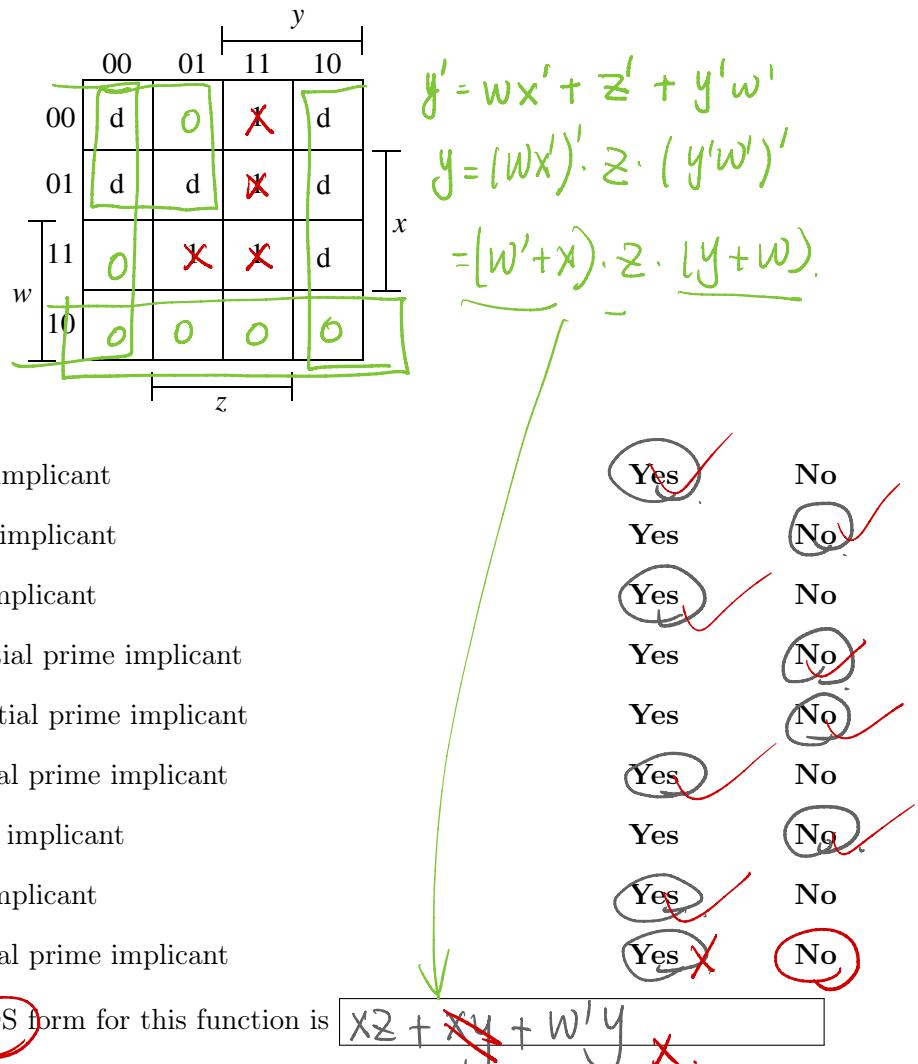
- b. [4 points] The figure below is the same as the one above except that the initial carry input is now 0 instead of 1. Determine the corresponding bit pattern on the circuit's outputs and write it out in the boxes below the respective output pins.

$$S = A \oplus B \oplus C_i$$

$$C_o = AB + (A \oplus B)C_i$$

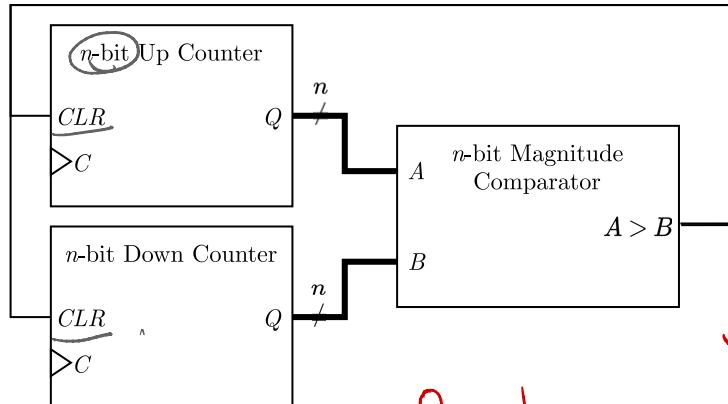


3 [K-Map Minimization–12 points]



4 [Counters–10 points]

This circuit below consists of two n -bit counters, one counting up and the other counting down, whose outputs are connected to an n -bit magnitude comparator. The result of the comparison is used to conditionally clear both counters, i.e., set them to 0; the counters' CLR inputs are synchronous with the clock. How long is the repeating counting sequence of either of the counters when:



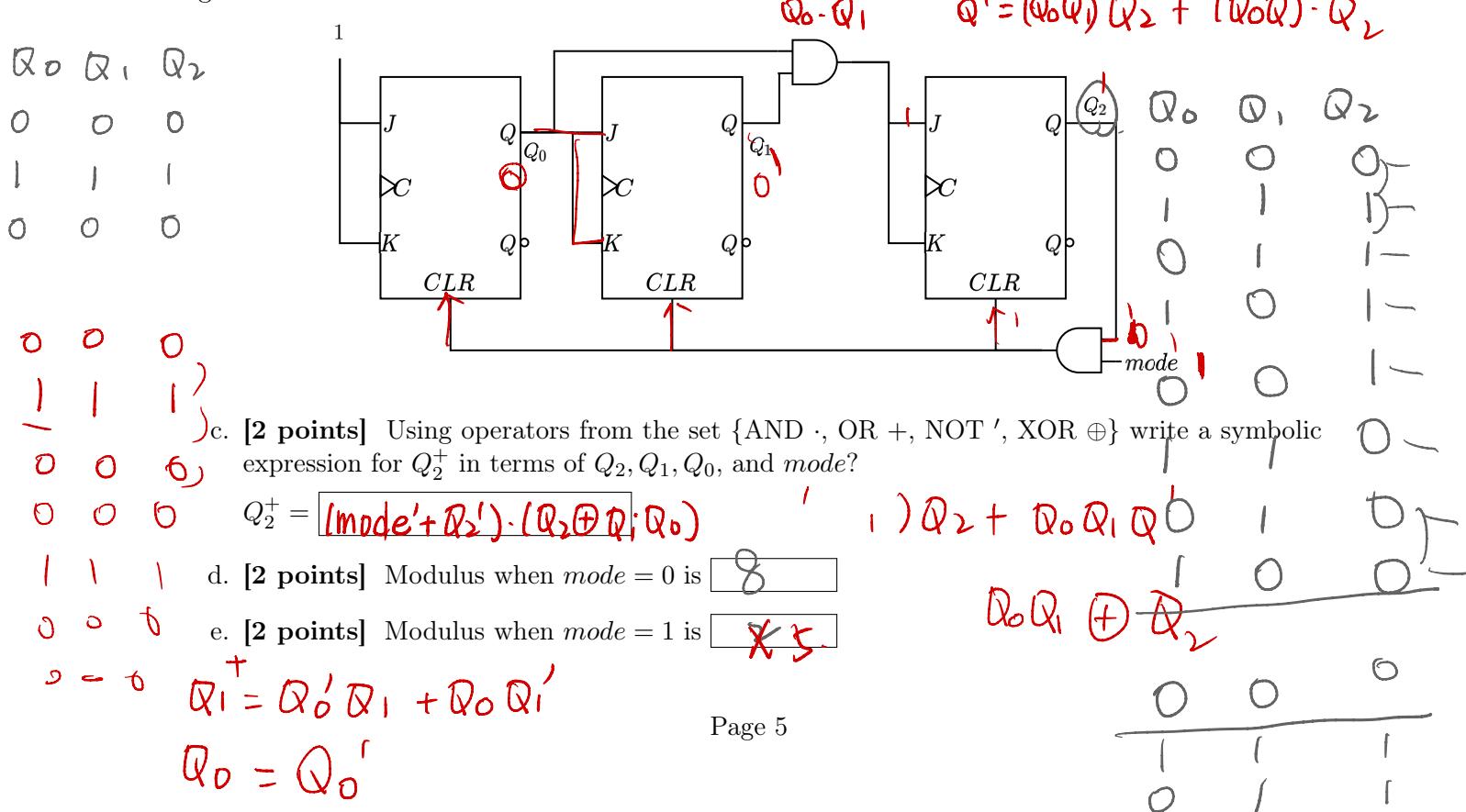
a. [2 points] $n = 3$:

b. [2 points] $n = 4$:

Q_2	Q_1	Q_0
0	0	0
1	1	1
0	0	0
0	0	0
1	1	1
0	0	0
0	0	0

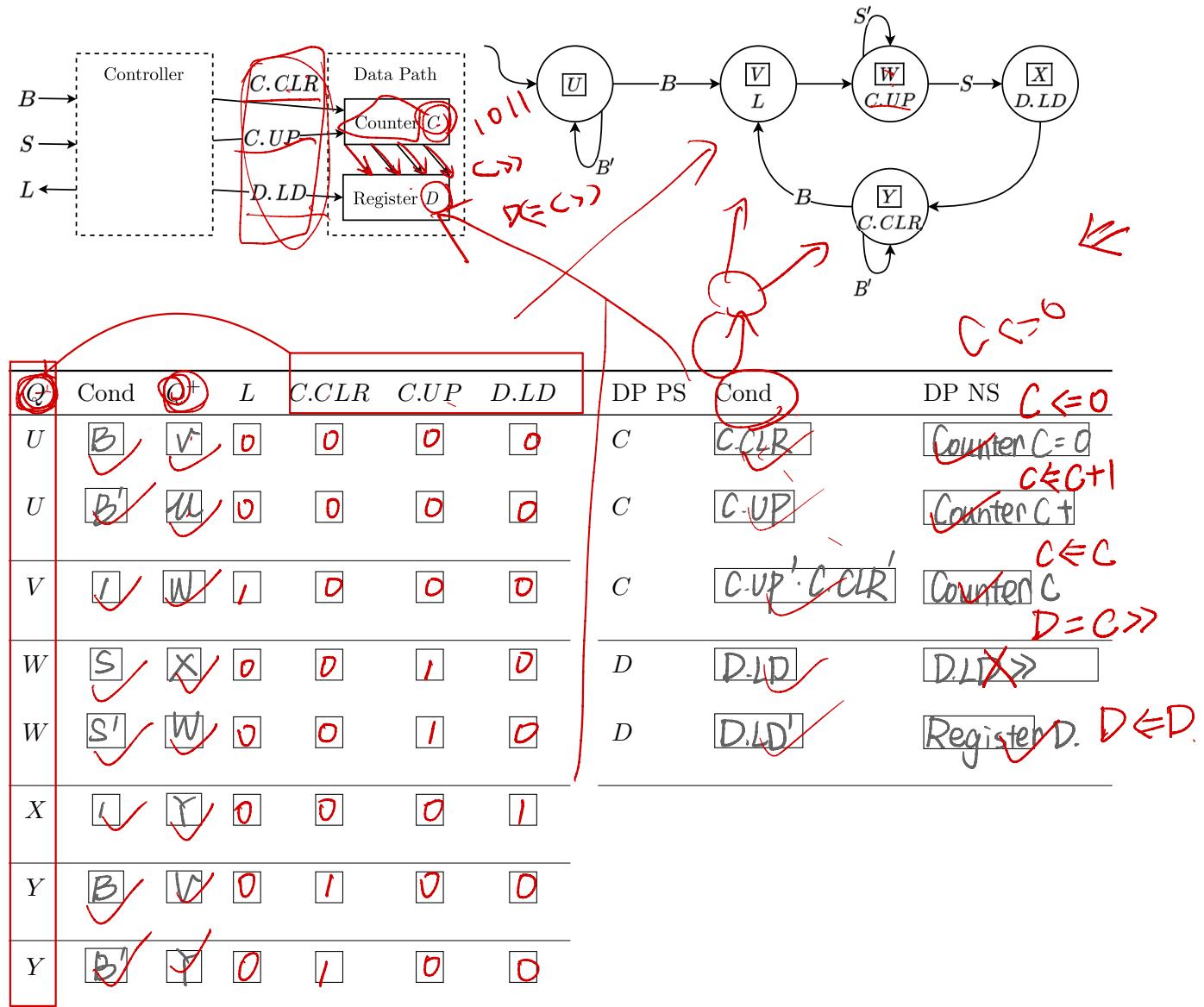
Q_2	Q_1	Q_0
0	0	0
1	1	1
0	1	0
1	0	1
0	1	0
1	0	1
0	1	0
1	0	1

The counter circuit below has a single *mode* input and uses 3 JK flip-flops with synchronous active-high clear.



5 [RTL Design–15 points]

The figure below shows the controller and datapath block diagram as well as the control state diagram for the laser measurer from homework # 8. Complete the transition lists for the controller and data path by entering in the boxes below Use $+$ for integer addition, \gg for right-shifting. Note that “Cond” is the symbolic transition expression (i.e., arrow label), and that “DP PS” and “DP NS” denote, respectively, the datapath present and next state.

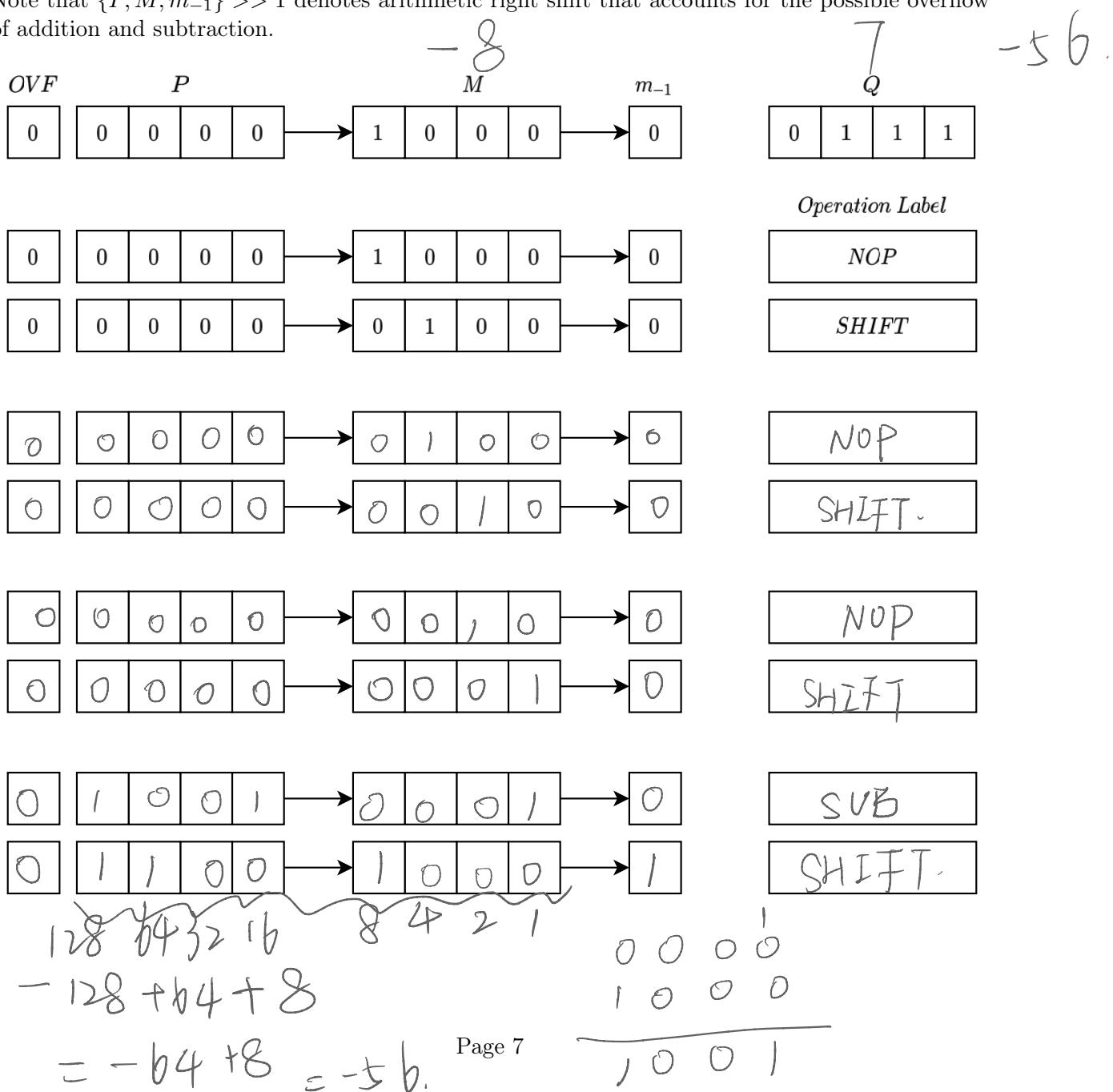


6 [Booth Multiplier—15 points]

The following trace shows the first iteration of the 4-bit Booth multiplication of -8×7 . Complete the trace for the remaining three iterations by entering the contents of the P and M registers as well as the OVF (overflow) bit and the m_{-1} flip-flop. Annotate the trace by indicating the operations performed in each iteration using the following “labels”:

Operation	Label
No operation	<i>NOP</i>
$P := P + Q$	<i>ADD</i>
$P := P - Q$	<i>SUB</i>
$\{P, M, m_{-1}\} >> 1$	<i>SHIFT</i>

Note that $\{P, M, m_{-1}\} >> 1$ denotes arithmetic right shift that accounts for the possible overflow of addition and subtraction.



7 [Carry Look Ahead–5 points]

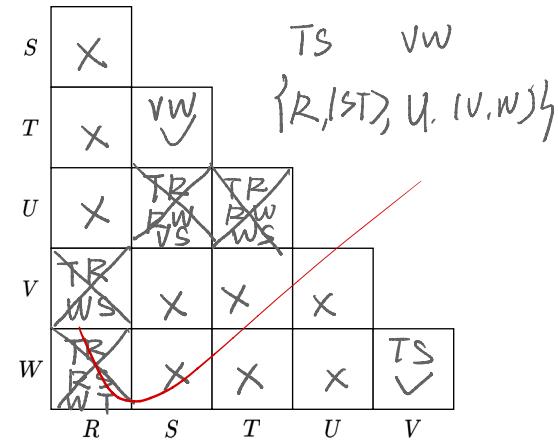
 Let $G_{j,i}$ and $P_{j,i}$ denote the group generate and propagate signals, respectively, for the bit range $[j, i]$ ($j \geq i$) in a carry look ahead adder. Note that the bit generate and bit propagate signals can be viewed as group signals over the single-bit range $[i, i]$, i.e., $g_i = G_{i,i}$ and $p_i = P_{i,i}$. Using this notation, which of the following is **not equal** to $G_{9,2}$?

- a. $G_{9,6} + P_{9,6} \cdot G_{5,2}$
- b. $G_{9,5} + P_{9,5} \cdot G_{4,2}$
- c. $G_{9,8} + P_{9,8} \cdot G_{7,2}$ \checkmark $G_{6,2} = G_{6,3} + P_{6,3} \cdot G_{2,2}$
- d. $G_{9,7} + P_{9,7} \cdot (G_{6,3} + P_{6,3} \cdot G_{2,2})$
- e. \checkmark They are all equal to $G_{9,2}$

8 [State Minimization–15 points]

The state table, shown below, is for a sequential circuit with two inputs X, Y and one output Z . The states are labeled R, S, T, U, V , and W . Complete the state merger diagram for this circuit and answer the following questions. Note: PS is Present State and NS is Next State.

PS	XY				Z
	00	01	10	11	
R	T	R	W	V	0
S	T	U	R	V	1
T	T	U	R	W	1
U	R	U	W	S	1
V	R	T	S	V	0
W	R	S	T	V	0
	NS				

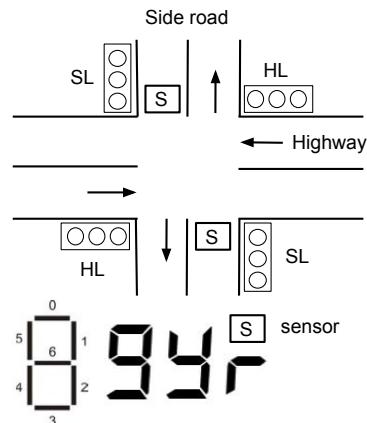


- a. [3 points] States W and S are equivalent True False
- b. [3 points] States T and S are equivalent True False
- c. [3 points] State V is equivalent to W
- d. [3 points] State U is equivalent to Nothing
- e. [3 points] The minimum number of states is 4

9 [Lab Experience—10 points]

Complete the following Verilog for a simple traffic signal with the following rules.

The highway light should be green for a minimum of 10 timer counts. The highway light should remain green as long as traffic is not detected on the side road. The side road light should remain green as long as the sensor is active, but not to exceed a timer count of 5. A yellow light transition occurs between green lights and lasts for just one FSM clock. Sensor is active high. HEX LEDs are active low.



```

module traffici(
    input clk, sensor, reset,
    output reg[6:0] HL, SL); // Highway and side street lights
    reg [1:0] [ ] ;
    reg [31:0] [ ] ;
    [ ] timer_reset;

    always@* begin
        case(s)
            0: if [ ] ns = 1; else ns = 0;//highway green state
            1: ns = 2;                      //highway to side yellow state
            2: if [ ] ns = 3; else ns = 2;//side street green state
            3: ns = 0;                      //side to highway yellow state
        endcase
    end

    always@(posedge clk) begin
        if([ ]) begin s <= 0; timer <= 0; end
        else [ ];
        if(timer_reset) timer <= 0;
        else [ ];
    end

    always@* begin
        case(s)
            0: begin HL = 7'b0010000; SL = 7'b0101111; timer_reset = 0; end
            1: begin HL = [ ]; SL = 7'b0101111; timer_reset = 1; end
            2: begin HL = 7'b0101111; SL = 7'b0010000; timer_reset = 0; end
            3: begin HL = [ ]; SL = 7'b0010001; timer_reset = 1; end
        endcase
    end
endmodule

```